Journal of Engineering Science and Technology
Vol. 17, No. 5 (2022) 3326 - 3341

AN INTELLIGENT ATTENDANCE SYSTEM BASED ON CONVOLUTIONAL NEURAL NETWORKS FOR REAL-TIME STUDENT FACE IDENTIFICATIONS

M. A. AHMED*, MOHANAD DAWOOD SALMAN, RAWAN ADEL ALSHARIDA, Z. T. AL-QAYSI, MAYTHAM M. HAMMOOD

Computer Science Department, College of Computer Science and Mathematics, Tikrit University (TU), Tikrit, Iraq
*Corresponding Author: mohamed.aktham@tu.edu.iq

Abstract

To date, managing student attendance is considered a challenging task in the higher education sector. Therefore, the technique of face recognition was recently adopted to cope with this problem. In this regard, a real-time student attendance system was developed to be deployed in small-scale environments such as lecture halls. To do this, three different techniques were used, such as face detection, identification, and counting. Face detection is accomplished using the Histogram of Oriented Gradients Algorithm. For face identification, a Convolution neural network was used to extract facial features. A Support Vector Machine was used to identify student faces in the image. Finally, a Haar cascade classifier was used to count the number of faces in the image during the counting step. As a result, the developed system achieved a classification accuracy of 99.75% in identifying faces in the captured image. The developed method can be considered an optimal solution to replace the traditional manual approach with an automated one. The developed system reduces the time-consuming paperwork. In addition, this automated system will provide an accurate result and save people's efforts for doing tedious work.

Keywords: Automatic system, Deep features, Face detection, Face recognition, Haar cascade, Histogram of oriented gradients, Support vector machine.

1. Introduction

One of the most basic requirements for measuring student engagement success in educational institutions is attendance monitoring [1]. Student attendance monitoring has become a big challenge for schools. Faculty must expend a significant amount of time and effort manually recording attendance [2]. In the traditional attendance manual approach, the attendance is taken either by calling students' names or bypassing the attendance sheet. Both approaches are time-consuming and may provide an opportunity for fraud [3]. Developing an automated attendance system could address the problems above [4].

Various attendance systems have been developed thus far using biometrics and non-biometrics method. Non-biometrics identification system uses mainly punch cards [5], Quick Response codes (Q.R.) [6], or Radio-frequency identification (RFID) to record student attendance [7]. QR-Code technology is identified by a barcode that can be read with a scanner. In contrast, RFID technology is identified by radio waves that can be read with a wave reader. The non-biometric recognition method has many benefits, including ease of use and high precision. However, the non-biometric recognition approach has flaws, such as the inability to complete the attendance process if students fail to bring object identification (OID). Furthermore, anyone with an OID can use it to record attendance [8].

Biometric identification-based attendance systems include fingerprint recognition [9], palm vein recognition [10], and iris recognition [11] based on physical and behavioural characteristics [12], and face recognition [13]. The main advantage of the biometric method is that physical features are still bound to the human. However, there are some deficiencies in several solutions. The fingerprinting approach is close to the traditional process of marking attendance [4]. Since students are required to scan their fingertips by a fingerprint scanner for a few seconds, this marks them present after a few seconds of processing [14]. The total time for the entire attendance marking process increases when we consider that each student would spend a few seconds, if not minutes, on the fingerprint scanner. All of the above is compatible with the iris scanning process of the retina [15]. As a result, applications based on fingerprints or iris scanners are often time-consuming [16].

Face recognition is one of the most challenging biometric modalities when deployed in unconstrained environments due to the high variability that faces images present in the real world [17]. Illumination conditions, occlusions, facial expressions, and Head positions are just a few of the variants [18]. Convolution Neural Networks (CNN) have recently addressed many computer vision problems, such as object detection and recognition, facial expression analysis, segmentation, age estimation, optical character recognition. CNN-based face recognition achieved very high accuracy. They can learn robust features to the real-world variations in the face images used during training [19]. To tackle the biometric identification-based techniques problems, CNNs technology in the form of deep features as input for machine face identification and enhance the identification accuracy can contribute to addressing the problem. Figure 1 shows the structure of the automatic student attendance system.

This article proposes biometric attendance monitoring by using an attendance registration system that uses CNN features and SVM-based face recognition as input data. Automatically marking student attendance and finally preparing and delivering reports to those concerned. Furthermore, the data augmentation technique is used in this study to address insufficient samples issues and minimise overfitting. The proposed attendance system based on face recognition technology is the ideal solution to these issues. Face recognition allows students' faces to be captured in real-time as they study in class without them knowing, resulting in a seamless learning experience.

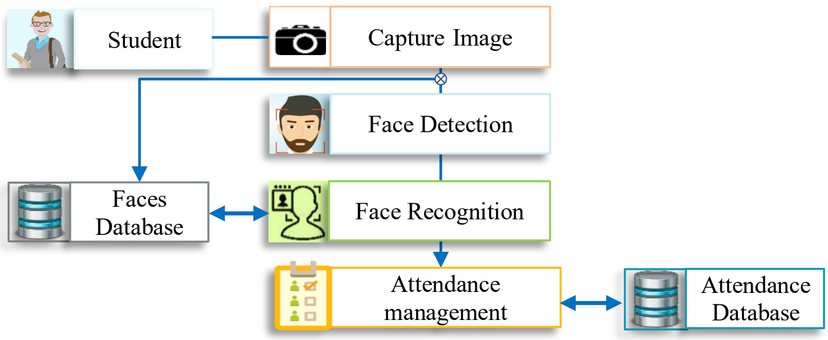

Fig. 1. Face-recognition-based attendance system steps.

The major contributions of this paper are summarised as follows:

- Provide the easiest and most effective solution for automatically registering attendance through face recognition by blending the deep feature extraction with SVM classification.
- The CNN with the SVM classification has been shown experimentally to achieve extremely excellent image classification efficiency.
- Face counting and face recognition are used together to improve system efficiency.
- The cost of the system has been minimised by using a single camera to take images in the lecture hall.

2. Literature Review

There are many successful cases of using automated attendance-taking systems. Sanli and Ilgen [20] proposed a camera-based system to capture student's images to record attendance automatically. The recognition process is achieved by Principal Component Analysis and Local Binary Pattern Histograms (LBPH) algorithms. The performance of the face recognition system was found to be 75%. However, it is also worth remembering that with some changes to the algorithms used, much better face recognition results in LBPH performance on such systems may be achieved.

Pei et al. [21] proposed a Deep Learning-based attendance-taking system. The VGG-16 was trained on 3538 photos of students' faces and then tested on 372 images. However, the training takes around 8 hours, which is a significant amount of time. Furthermore, the recognition accuracy of the proposed method reached 86.3%, which is not high. Indra et al. [22] presented the Haar-like features technique for developing student attendance systems utilising facial recognition patterns.

Khan et al. [23 proposed an attendance monitoring system based on face recognition. The device uses the YOLO V3 (You Only Look Once) algorithm for face detection. However, its flaw is that it is prolonged if implemented on the CPU. The system is not suitable for face identification in real-time. Using the Deep Convolutional Neural Network approach, Setialana et al. [24] developed an intelligent attendance system. The system can accurately record student attendance with an accuracy of 81.25 per cent. However, the system's accuracy was not evaluated in the study in varied lighting situations or with different camera quality.

Shah et al. [25] proposed an Automatic Attendance System based on face recognition that identifies students' faces. The system achieved 93.1% accuracy for student face recognition. On the other hand, noise in images significantly impacts recognition performance, resulting in a loss of overall accuracy.

According to the discussion above, the accuracy achieved of proposed attendance systems is insufficient for a reliable system in real-time student identification. Thus, the prominent issue related to recognition accuracy should be addressed. In this research, we present a solution that can handle the problems mentioned above. We will use deep features and SVM for real-time student face identifications to automatically mark attendance, saving time and reliability.

3. Material and Method

Algorithm 1 explained the main steps of the proposed intelligent attendance monitoring system. Figure 2 depicts the different phases of the life cycle of system deployment.

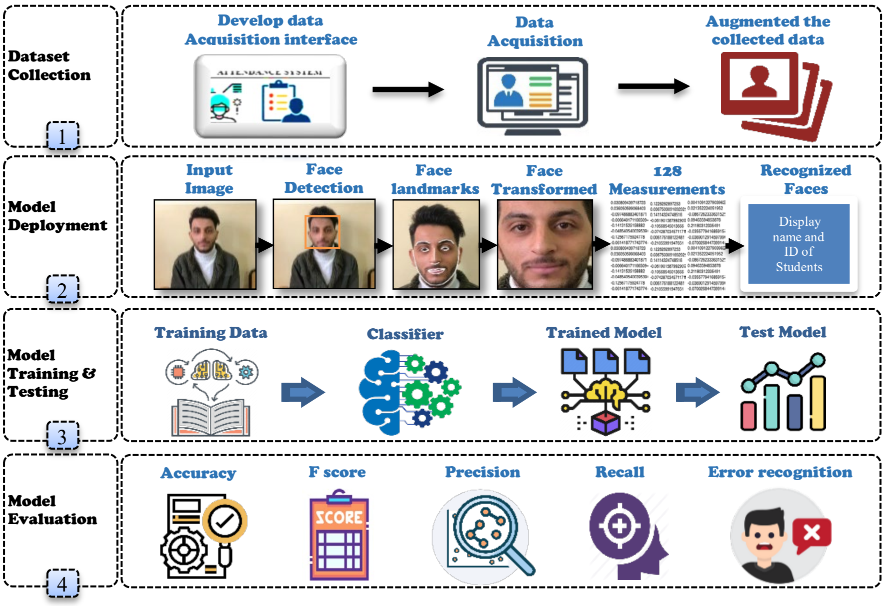

Fig. 2. The life cycle of an intelligent attendance monitoring system.

Algorithm 1. Pseudo Code of student identification

Capturing_student Image by camera

Input : Student_image_file

```
    Output: Update the database
    N: Number_identified_faces
    M: Count_ Faces
1.  Convert: image → im_gray
2.  HOG_ (im_gray) →Face_Detection
3.  F_landmark_est (Face _Detection, 68) → find 68 specific points
4.  Draw: plot_ROI→ rectangle_bounding_box
5.  Encoded:  im_ROI → feature_vector(128)
6.  SVM_classifier→N
7.  Haar_cascade_classifier →M
8.  If N = M then
9.  |  Display: names, IDs
10. Else if trial_number < 6
11. |  trial_number ++
12. |  Repeat steps: 1 to 9
13. Else
14. |  Display: error message
15. End if
```

3.1. Dataset creation

Algorithm 2 shows the procedures proposed in this research for data collection. Ten photos are taken for each student in the perpendicular direction with different facial expressions (i.e., normal, smiling, anger, and disgust), different look (i.e., wearing glasses or not), different side view, and different distance. Eight students volunteered in this research. The ten captured images per student are stored in a folder with a unique name that includes the student's I.D., and all images are in PNG format. A data augmentation technique was applied during this stage to alleviate model overfitting and improve the model's generalisation ability [26]. For this purpose, "ImageDataGenerator," the Keras API, is used for doubling the size of data. In the augmentation process, each image is transformed into a 20 ° range of rotation. In addition, the zoom is set to 20%, the shear was set to 20%, and eventually, we retain the true horizontal flip, noising, and blurring.

Algorithm 2: Dataset preparation

```
1-  Capturing_student Image by camera
    Input  : Student_image_file , N times
    Output: Update the database
2-  for j=0 to N times
3-  |  Save image → Disk
4-  |  Load image_Disk → stImage
5-  |  for i=0 to length(stImage)
6-  |  |  stImage0=Shift_transform(stImage)
7-  |  |  stImage1=Flip_transform(stImage0)
8-  |  |  stImage2=Rotation_transform(stImage1)
9-  |  |  stImage3=Zoom_Transform(stImage2)
10- |  End for
11- End for
12- Save stImage → Disk
13- Upload → Database
```

In the server, the data augmentation technique is applied to the uploaded images. For each image, 45 transformed versions of original images are created using the data augmentation technique. Accordingly, the face image data set consisted of 3600 (8×10×45) images created and used to build a classifier for face recognition. Figure 3 shows an example of face images in the data set. The augmented images are shown in Fig. 4. The authors have received permission from students whose face images are used in this paper.

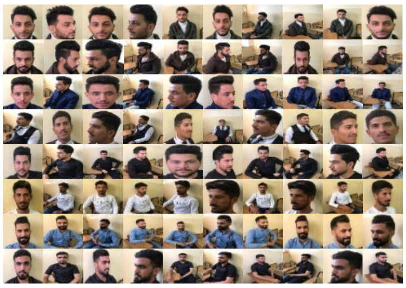

Fig. 3. The example of the student image in the dataset.

Fig. 4. The example of the augmented images in the dataset.

3.2. Model deployment

The face recognition module consists of several machine-learning processes: face detection, finding face landmarks and positioning, face identification.

3.2.1. Face detection

Histograms of Oriented Gradient (HOG) method is used to locate the faces in an image. HOG is a feature descriptor that is often used to extract features from image data. It is widely used in computer vision tasks for object detection. HOG is an adequate descriptor for object recognition in general and faces recognition in particular [27]. HOG algorithm work to determine the darkness of the specific pixel compared to the pixels directly surrounding it. Use this information to draw an arrow in the direction in which the pixel becomes darker. Face detection using the HOG method is shown in Algorithm 3.

Algorithm 3. Pseudo Code of a Histograms of Oriented Gradient Algorithm

```
Input: the original image
Output: image with face indicators as a rectangle
Ns : num of scales in the pyramid of images do
Nsh: num of shift steps of sub-window do
Nst : num of stages in cascade, classifier do
Nf : num of filters of stages k do
create_ imagei (down_sample)
For i = 1 to Ns
  compute: integral_image, imageii
  For j= 1 to Nsh
    For k=1 to Nst
      For l=1 to Nf
        Filter_detection(sub-window)
        Filter_accumulate → outputs
      End for
      if accumulation/stage threshold = false then
        Reject → sub-window _face
        break k → loop:
```

```
        End if
      End for
      if sub-window/ stage_checks = True then
      |sub-window → accept_face
      End if
    End for
End for
```

3.2.2. Face landmarks

The face landmark estimation algorithm is used to extract features from images for the same person. The algorithm produces 68 specific points (called landmarks) such as the chin, the inner edge eyebrow, the outer edge of the eye. These landmarks are present on every human face and help to determine the position of eyes and mouth. Algorithm 4 illustrates the steps of the landmark estimation method [28].

Algorithm 4. Pseudo Code of face landmark estimation Algorithm

Input: image with face indicators as a rectangle

Output: image with 68 facial landmarks

Have training data $\left\{\left(I_{\pi_i}, \hat{\mathbf{S}}_i^{(t)}, \Delta\mathbf{S}_i^{(t)}\right)\right\}_{i=1}^{N}$ and the learning rate (shrinkage factor) $0 < v < 1$

1. Initialise

$$f_0\left(I, \hat{\mathbf{S}}^{(t)}\right) = \arg\min_{\gamma \in \mathbb{R}^{2p}} \sum_{i=1}^{N} \left\|\Delta\mathbf{S}_i^{(t)} - \gamma\right\|^2$$

2. *for* $k = 1, \ldots, K$:

   (a) Set *for* $i = 1, \ldots, N$

$$\mathbf{r}_{ik} = \Delta\mathbf{S}_i^{(t)} - f_{k-1}\left(I_{\pi_i}, \hat{\mathbf{S}}_i^{(t)}\right)$$

   (b) Fit a regression tree to the targets rik giving a weak regression function $g_k\left(I, \hat{\mathbf{S}}^{(t)}\right)$.

   (c) Update

$$f_k\left(I, \hat{\mathbf{S}}^{(t)}\right) = f_{k-1}\left(I, \hat{\mathbf{S}}^{(t)}\right) + v g_k\left(I, \hat{\mathbf{S}}^{(t)}\right)$$

3. Calculate regression function $r_t\left(I, \hat{\mathbf{S}}^{(t)}\right) = f_K\left(I, \hat{\mathbf{S}}^{(t)}\right)$
4. Warp the image to the mean shape based on the current shape estimate for shape invariant split

$$h\left(I_{\pi_i}, \hat{\mathbf{S}}_i^{(t)}, \boldsymbol{\theta}\right) = \begin{cases} 1 & I_{\pi_i}(\mathbf{u}') - I_{\pi_i}(\mathbf{v}') > \tau \\ 0 & \text{otherwise} \end{cases}$$

   Each split is a decision involving 3 parameters θ = (τ, u, v).

5. Choosing the node splits

   (a) minimizes the sum of square error

$$E(\mathcal{Q}, \boldsymbol{\theta}) = \sum_{s \in \{l,r\}} \sum_{i \in \mathcal{Q}_{\boldsymbol{\theta},s}} \|\mathbf{r}_i - \boldsymbol{\mu}_{\boldsymbol{\theta},s}\|^2$$

   (b) calculated from the average of the targets at the parent node μ and μθ,l

$$\boldsymbol{\mu}_{\boldsymbol{\theta},r} = \frac{|\mathcal{Q}|\boldsymbol{\mu} - |\mathcal{Q}_{\theta,l}|\boldsymbol{\mu}_{\boldsymbol{\theta},l}}{\mathcal{Q}_{\boldsymbol{\theta},r}}$$

6. Output $P(\mathbf{u}, \mathbf{v}) \propto e^{-\lambda\|\mathbf{u}-\mathbf{v}\|}$

3.2.3. Face identification

The human face recognition algorithm has three steps: feature extraction, training, and database classification.

A. Feature extraction

FaceNet is used for feature extraction in this research due to its high performance and less extraction time than other CNN models [29]. In the FaceNet model, the triplet loss function is chosen to use. Triplet loss can learn from a match between the anchor photo and the positive example and between the anchor and negative. In general, feature extraction using the FaceNet model, as seen in Fig. 5, involves feeding input images into a deep learning architecture, which is then normalised L2, yielding facial features (embedding) trained using the Triplet Loss function. Table 1 depicts the architecture of the FaceNet model.

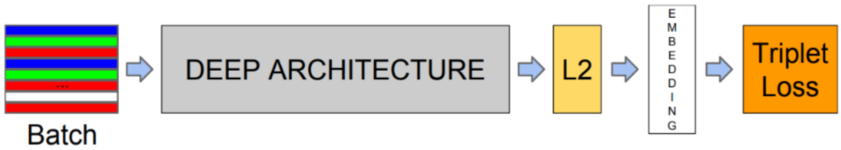

Fig. 5. Feature extraction using Face Net [30].

Table 1. Description of the CNN architecture used for feature extraction.

Layer Type	Output	Size Depth	Pool	Params
conv1 (7 ×7 ×3,2)	112 ×112 ×64	1		9 K
max pool + norm	56 ×56 ×64	0	m 3 ×3, 2	
inception (2)	56 ×56 ×192	2		115 K
norm + max pool	28 ×28 ×192	0	m 3 ×3, 2	
inception (3a)	28 ×28 ×256	2	32 m, 32p	164 K
inception (3b)	28 ×28 ×320	2	L2, 64p	228 K
inception (3c)	14 ×14 ×640	2	m 3 ×3,2	398 K
inception (4a)	14 ×14 ×640	2	L2, 128p	545 K
inception (4b)	14 ×14 ×640	2	L2, 128p	595 K
inception (4c)	14 ×14 ×640	2	L2, 128p	654 K
inception (4d)	14 ×14 ×640	2	L2, 128p	722 K
inception (4e)	7 ×7 ×1024	2	m 3 ×3,2	717 K
inception (5a)	7 ×7 ×1024	2	L2, 128p	1.6 M
inception (5b)	7 ×7 ×1024	2	m, 128p	1.6 M
avg pool	1 ×1 ×1024	0		
fully conn	1 ×1 ×128	1		131 K
L2 normalization	1 ×1 ×128	0		

B. Classification

SVM was used to classify the dataset in this research. The Support Vector Machine (SVM) is an efficient classifier [26, 31]. SVM is a fast machine learning algorithm for multi-class classification that can be applied to large data sets. SVM carries out data classification by forming an optimum N-dimensional hyperplane, splitting data samples into positive and negative. SVM has a kernel and two critical parameters, namely cost, c and gamma, γ. The cost parameter regulates the trade-off between acquiring accuracy in training and accuracy in testing. At the same time, the gamma parameter governs the effect of a single data sample of training, given the feature vector xi, weight vector w, and class label yi [32]. SVR problem can be formulated as shown in Eq. (1) and Subject to Eq. (2) [33, 34].

$$\text{Minimise:} \frac{1}{2} \parallel w \parallel + C\left(\sum_i \xi_i\right) \tag{1}$$

Subject to : $y_i(w^T x_i + w_0) \geq 1$ (2)

where i represents the no. of samples, C is the cost parameter controlling the maximisation of margin and minimisation of classification error, and ξ represents the training errors.

C. Train and test data

In this study, the split data approach is used to evaluate the performance of the recognition method. The original samples used to construct the dataset are 3600 face images divided into eight categories. The dataset is normalised to speed uploading during preparation, and the training data is supplemented during the CNN training phase. For training and testing, the dataset was split into 80% and 20%. This study used the confusion matrix and accuracy metric to assess the proposed model in terms of performance. Table 2 shows the train and test data in depth.

Table 2. Details of train and test data.

	Percentage	#Samples
Train	80	2880
Test	20	720

3.3. Face counting

Haar feature-based cascade classifier is used to count the people in an image. Many objects counting algorithms are highly efficient, but their long processing times prevent them from being used for real-time detection. On the other hand, the Haar-cascade algorithm is an ML-based technique [27]. As seen in Algorithm 5, the algorithm has four stages: Haar feature selection, integral image development, AdaBoost training, and cascading classifiers.

Algorithm 5. Pseudo Code of Haar-cascade algorithm

1. **Input**: the image
2. **Output**: number of faces in a given image
3. Pick f (maximum acceptable false-positive rate per layer) and d (minimum acceptable false-positive rate per layer)
4. Lets $F_{tareget}$ is target overall false-positive rate
5. Lets P is a set of positive examples
6. Lets N is set of negative examples
7. Lets F_0=1, D_0=1, and i=0 (F_0: overall false positive rate at layer 0, D_0: acceptable detection rate at layer 0, and i: is the current layer)
8. While F_i>F_{target}
9. i++ (layer increasing by 1)
10. n_i=0; $F_i = F_i$ -1(ni:negative example i):
11.   while F_i >f* F_i -1
12. n_i++(check next negative example)
13.   use P and N to train the AdaBoost to make xml (classifier)
14.   check the result of new classifier for F_i and D_0
15.   decrease threshold for new classifier to adjust detection rate r>=d* F_i -1
16. N=empty
17. *If* F_i >F_{target}, use the current classifier and false detection to set N

4. Application Module

The Application Module contains three Android applications that differ in their characteristics: the teacher application, the student application, and the family application. By teacher application, the instructor inputs the teaching code in terms of the date and time of the course and takes a photo of students with the mobile camera and the image is uploaded to a server. By student application, the student fills in his private information such as name, I.D., stage, and email. In addition, the student receives a notification of the attendance-check result for each course and gets a warning when the limit of acceptable absence is exceeded. By attendance processing, the result of the face recognition module is passed to the attendance-monitoring unit, which is recorded the attendance electronically. In the reports unit, student attendance reports are created by accessing the SQL Server database. This program is responsible for sending notifications and alerts to the student, family, and administration by email or SMS per student.

5. Result and Discussion

Two datasets used in this study: the original dataset contains 80 samples, and the augmented dataset contains 3600 samples from eight different subjects. The 128 embedded features were used as input to the SVM classifier for each sample. Performance metrics were used to calculate the performance of the classifier model for the training and testing phases. The findings were organised into several tables. The overall recognition accuracy of each dataset was highlighted based on the results reported in the tables. The training- and test-based comparative analysis results for the datasets are presented in Tables 3and 4.

Table 3. Training results of face recognition using SVM classifier.

Dataset	Accuracy	Precision	Recall	F1-score
Original Dataset	0.98	0.98	0.97	0.98
Augmented Dataset	0.99	0.99	0.98	0.99

Table 4. Testing face recognition results using an SVM classifier.

Dataset	Accuracy	Precision	Recall	F1-score
Original Dataset	0.97	0.98	0.98	0.97
Augmented Dataset	0.99	0.99	0.99	0.99

Almost all two datasets presented high training accuracy, as shown in Table 3. The augmented dataset provided the highest accuracy of approximately 99% for face recognition. The test results of the face recognition using SVM have an accuracy ranging between 99% and 97%. The identification accuracy with the augmented dataset was approximately 99%, as shown in Table 4. Table 5 presented the accuracy of each class.

Table 5. The accuracy was obtained using original and augmented datasets per class.

Class	Acc of original	Acc of augmented	Class	Acc of original	Acc of augmented
STU_ID1	99%	100%	**STU_ID5**	100%	100%
STU_ID2	100%	100%	**STU_ID6**	97%	99%
STU_ID3	98%	99%	**STU_ID7**	98%	100%
STU_ID4	99%	100%	**STU_ID8**	96%	100%

The per-class accuracy data showed promising results for most classes. Four classes yield up to 99% accuracy with the original dataset. For the augmented dataset, almost all classes had an accuracy of 100 %, and two had accuracies between 99% (Table 4).

Two hundred images with different people were captured using a smartphone camera to evaluate the model in a real case study (see Table 6). The flowchart in Fig. 6 demonstrates the approach used to measure the system performance metrics. The accuracy is calculated using Eq. (3) [35], and error rates are calculated using Eq. (4) [36] as listed below:

$$Accuracy\% = \frac{\text{faces detected right}}{no.\text{ of faces in an image}} \times 100 \tag{3}$$

$$Err\ wrong\ detected = \frac{\text{faces detected wrong}}{\text{no. of faces in an image}} \times 100 \tag{4}$$

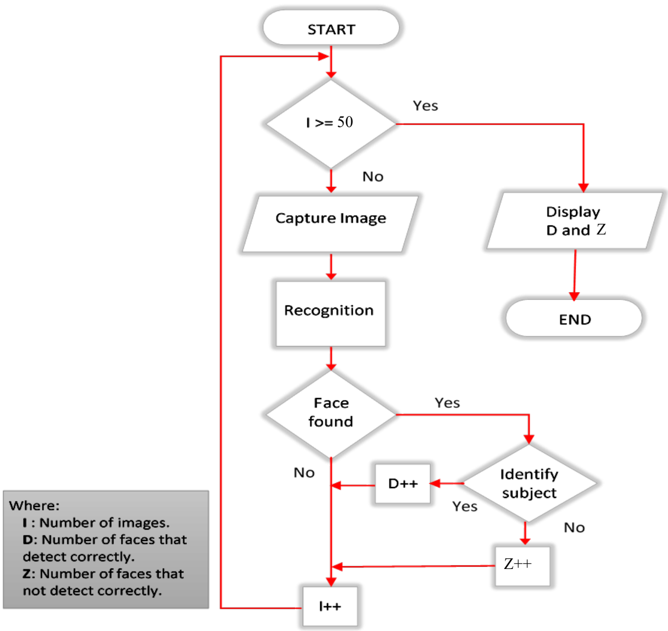

Fig. 6. Image testing procedure flowchart [37].

Table 6. Accuracy and error values of detected faces in each image.

Number of people in the image	Number of test image	Number of properly recognized images	Accuracy
8	50	50	100%
15	50	50	100%
20	50	47	94%
30	50	46	92%
All images	200	193	96.5%

By analysing Table 6, when the number of people in a picture is between 8 and 15, the accuracy of facial recognition reaches 100%. The table also reveals that as the number of faces in an image grows, correctly recognised images decline. Since examining the images, it was noticed that some of the images for students involve a shadow, reflection, which could lead to misclassification. Similarly, a picture of a blurred or occluded (i.e., a Veiled face by a barrier) face leads to an undetected mistake. In some instances, these issues are the reason the faces are not correctly recognised. The total accuracy of the student recognition system reached 96.5%. Figure 7 depicts the system's output in Face Detection and Recognition for images taken from various perspectives.

Fig. 7. Face detection and recognition result of pictures taken with different views.

In Table 7, studies based on face recognition for mark student attendance are used as an alternative for benchmarking purposes. The results of this work are compared with the benchmark studies to study the efficiency of the proposed intelligent attendance management system. The criterion for evaluation is accuracy.

Table 7. Comparison of our method versus the related studies.

Reference	Student no	Dataset samples	Face recognition method	Matching method	Accuracy
[20]	10	100	Principal Component Analysis and Local Binary Pattern Histograms (LBPH) algorithms	non	75%.
[21]		3563	VGG-16	non	86.3%,
[22]	20	100	Bilateral Filter, and Haar-like features	non	
[23]	12	240	YOLO V3	non	
[24]	16	150	CNN	non	81.25%
[25]	6	1200	Haar Cascade classifier	non	93.1%
Our proposed	8	3600	Deep CNN features & SVM	Haar-cascade algorithm	99.75%

According to Table 7, it is clear that most of the previous research works were conducted on a limited number of images for training and testing the recognition models. Among studies, the proposed method in this study yielded the highest accuracy reach to 99.75%. Furthermore, this approach is reliable since the result compares face recognition and face counting results.

6. Limitations

The proposed automatic student attendance monitoring system is often found to have certain limitations. When the number of images in the database is considerable, the processing time needed is significant. In the real-time situation, more analysis should be done to improve the system's performance. Additionally, the number of training images per student can be increased to make the system more resistant to the recognition challenge. Furthermore, various methods for reducing computational time can be investigated and accomplished using far more effective algorithms.

7. Conclusions

An intelligent attendance recording, and management system is proposed based on existing face recognition algorithms. The method has been tested in various scenarios. In the vast majority of circumstances, it has shown to be 99.75 % accurate. The resulting system has fulfilled the study's initial purpose: to create an effective recording and recording of attending lectures. The experimental findings demonstrate the efficacy of the proposed method for student identification. Thus, this system could be used in any institution to monitor student and staff attendance. Accordingly, this device saves the lecturer time and manual labour that would otherwise be needed. Furthermore, this system assists lecturers in effectively managing large groups of students in the classroom.

Hopefully, this research will meet the rising demand for technological progress in the attendance-taking process, as the existing method has several problems that must be addressed. Some issues, however, will need to be discussed in the future. For instance, background dynamics, in which the system recognises the face despite the background moving. Other concerns will include changes to the student's appearance and the illumination of the setting, which must be sufficiently lit for face recognition.

Acknowledgement

We would like to thank the volunteers who willingly gave their images for this study.

Nomenclatures

2DFLD	Fisher's Linear Discriminate
API	Application Programming Interface
CNN	Convolutional Neural Network
HOG	Histogram of Oriented Gradients
IFRS	Indexed Face Recognition System
LBPH	Local Binary Pattern Histograms
LMS	Learning Management System
OID	Object Identification

PCA	Principal Component Analysis
QR	Quick Response Codes
RFID	Radio-Frequency Identification
ROI	Region of Interest
SVM	Support Vector Machine
YOLO V3	You Only Look Once

References

1. Sutabri, T.; Pamungkur, A. K.; and Saragih, R.E. (2019). Automatic attendance system for university student using face recognition based on deep learning. *International Journal of Machine Learning and Computing*, 9(5), 668-674.
2. Surekha, B.; Nazare, K.J.; Raju, S.V.; and Dey, N. (2017). Attendance recording system using partial face recognition algorithm. In Dey, N., Santhi, V. (eds.) *Intelligent techniques in signal processing for multimedia security*, 660, 293-319.
3. Krishnan, M.G.; Balaji; and Babu, S. (2015) Implementation of automated attendance system using face recognition. *International Journal of Scientific & Engineering Research*, 6(3), 30-33.
4. Mohammed, K.; Tolba, A.S.; and Elmogy, M. (2018). Multimodal student attendance management system (MSAMS). *Ain Shams Engineering Journal* 9(4), 2917-2929.
5. Khan, M.B.; Prashanth, N.M.; Nomula, N.; Pathak, P.; and Muralidhar, A.M. (2017). Auto student attendance system using student ID card via Wi-Fi. *Project Work*, Department of Computer Science and Engineering University of Bridgeport, Bridgeport, CT.
6. Wei, X.; Manori, A.; Devnath, N.; Pasi, N.; and Kumar, V. (2017). QR code based smart attendance system. *International Journal of Smart Business and Technology*, 5(1), 1-10.
7. Pss, S.; and Bhaskar, M. (2016). RFID and pose invariant face verification based automated classroom attendance system. In 2016 *International Conference on Microelectronics*, *Computing and Communications* (*MicroCom*), Durgapur, India, 1-6.
8. Poornima, S.; Sripriya, N.; Vijayalakshmi, B.; and Vishnupriya, P. (2017). Attendance monitoring system using facial recognition with audio output and gender classification. In 2017 *International Conference on Computer*, *Communication and Signal Processing* (*ICCCSP*), Chennai, India, 1-5.
9. Mohamed, B.K.P.; and Raghu, C.V. (2012). Fingerprint attendance system for classroom needs. In 2012 *Annual IEEE India Conference* (*INDICON*), Kochi, India, 433-438.
10. Bayoumi, S.; Aldayel, A.; Alotaibi, M.; Aldraihem, M.; Alrashed, S.; and Alzahrahi, S. (2015). Class attendance system based-on palm vein as biometric information. *Journal of Theoretical and Applied Information Technology*, 77(2), 266-272.
11. Okokpujie, K.O.; Noma-Osaghae, E.; Okesola, O.J.; John, S.N.; and Robert, O. (2017). Design and implementation of a student attendance system using iris biometric recognition. In 2017 *International Conference on Computational Science and Computational Intelligence* (*CSCI*), Las Vegas, NV, US, 563-567.

12. Shawkat, S.A.; and Ismail, R.N. (2019). Biometric technologies in recognition systems: A survey. *Tikrit Journal of Pure Science*, 24(6), 132-137.
13. Elias, S.J.; Hatim, S.M.; Hassan, N.A.; Abd Latif, L.M.; Ahmad, R.B.; Darus, M.Y.; and Shahuddin, A.Z. (2019). Face recognition attendance system using local binary pattern (LBP). *Bulletin of Electrical Engineering and Informatics* 8(1), 239-245.
14. Agrawal, A.; Garg, M.; Prakash, S.; Joshi, P.; and Srivastava, A.M. (2020). Hand down, face up: Innovative mobile attendance system using face recognition deep learning. In *Proceedings of* 3*rd International Conference on Computer Vision and Image Processing*, 363-375.
15. Bah, S.M.; and Ming, F. (2020). An improved face recognition algorithm and its application in attendance management system. *Array*, 5, 100014.
16. Sunaryono, D.; Siswantoro, J.; and Anggoro, R. (2019). An android based course attendance system using face recognition. *Journal of King Saud University - Computer and Information Sciences*, 33(3), 304-312.
17. Tejas, G.S.; Surya, G.; Supreetha, H.; Zaiban, S.; and Harish, K. (2021). A Survey on automated student attendance management system using face recognition. *Perspectives in Communication, Embedded-Systems and Signal-Processing (PiCES) – An International Journal*, 5(2), 19-21.
18. Vasquez, S.R. (2020). Facial recognition: Traditional methods vs. methods based on deep learning. In *International Conference on Information Technology & Systems*, 615-625.
19. Boucherit, I.; Zmirli, M.O.; Hentabli, H.; and Rosdi. B.A. (2020). Finger vein identification using deeply-fused convolutional neural network. *Journal of King Saud University - Computer and Information Sciences*, 34(3), 646-656.
20. Sanli, O.; and Ilgen, B. (2018). Face detection and recognition for automatic attendance system. In *Proceedings of SAI Intelligent Systems Conference*, 237-245.
21. Pei, Z.; Xu, H.; Zhang, Y.; Guo, M.; and Yang, Y.-H. (2019). Face recognition via deep learning using data augmentation based on orthogonal experiments. *Electronic*, 8(10), 1088.
22. Indra, E.; Yasir, M.; Andrian, A.; Sitanggang, D.; Sihombing, O.; Tamba, S.P.; and Sagala, E. (2020). Design and implementation of student attendance system based on face recognition by haar-like features methods. In 2020 3*rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT)*, Medan, Indonesia, 336-342.
23. Khan, S.; Akram, A.; and Usman, N. (2020). Real time automatic attendance system for face recognition using face API and OpenCV. *Wireless Personal Communications*, 113, 469-480.
24. Nurkhamid; Setialana, P.; Jati, H.; Wardani, R.; Indrihapsari, Y.; and Norwawi, N.M. (2021). Intelligent attendance system with face recognition using the deep convolutional neural network method. *Journal of Physics: Conference Series*, 1737 012031.
25. Shah, K.; Bhandare, D.; and Bhirud, S. (2021). Face recognition-based automated attendance system. In *International Conference on Innovative Computing and Communications*, 945-952.
26. Abdulateef, S.K.; Abdali, T.-A.N.; Alroomi, M.D.S.; and Altaha, M.A.A. (2020). An optimise ELM by league championship algorithm based on food

images. *Indonesian Journal of Electrical Engineering and Computer Science*, 20(1), 132-137.

27. Déniz, O.; Bueno, G.; Salido, J.; and De la Torre, F. (2011). Face recognition using histograms of oriented gradients. *Pattern Recognition Letters* 32(12), 1598-1603.
28. Kazemi, V.; and Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. *Proceedings of the* 2014 *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 1867-1874.
29. Agarwal, R.; Jain, R.; Regunathan, R.; and Kumar, C.S.P. (2019). Automatic attendance system using face recognition technique. In *Proceedings of the* 2*nd International Conference on Data Engineering and Communication Technology*, 525-533.
30. Taniai, H. (2018). keras-facenet. Retrieved October 5, 2021, from https://github.com/nyoki-mtl/keras-facenet
31. Sun, W.; and Ma, G.-Z. (2009). Safety assessment in power supply enterprise based on kernel principal component analysis and fast multi-class support vector machine. In 2009 *International Conference on Machine Learning and Cybernetics*, Hebei, 1214-1218.
32. Ahmad, M.A. (2016). Artificial neural network vs. support vector machine for speech emotion recognition. *Tikrit Journal of Pure Science*, 21(6), 167-172.
33. Misra, D.; Oommen, T.; Agarwal, A.; Mishra, S.K.; and Thompson, A.M. (2009). Application and analysis of support vector machine based simulation for runoff and sediment yield. *Biosystems Engineering*, 103(4), 527-535.
34. Welling, M. (2004). Support vector regression. Department of Computer Science, University of Toronto, Toronto, Canada.
35. Ahmed, M.A.; Zaidan, B.B.; Zaidan, A.A.; Salih, M.M.; Al-qaysi, Z.T.; and Alamoodi, A.H. (2021). Based on wearable sensory device in 3D-printed humanoid: A new real-time sign language recognition system. *Measurement*, 168, 108431.
36. Yang, H.; and Han, X. (2020). Face recognition attendance system based on real-time video processing. *IEEE Access*, 8, 159143-159150.
37. Ahmed, M.A.; Zaidan, B.B.; Zaidan, A.A.; Alamoodi, A.H.; Albahri, O.S.; Al-Qaysi, Z.T. ; Albahri, A.S.; and Salih, M.M. (2021). Real-time sign language framework based on wearable device: analysis of MSL, DataGlove, and gesture recognition. *Soft Computing*, 25, 11101-11122.